实验四

代码：#include<string.h>  
 #include<ctype.h>  
 #include<malloc.h> /\* malloc()等 \*/  
 #include<limits.h> /\* INT\_MAX等 \*/  
 #include<stdio.h> /\* EOF(=^Z或F6),NULL \*/  
 #include<stdlib.h> /\* atoi() \*/  
 #include<io.h> /\* eof() \*/  
 #include<math.h> /\* floor(),ceil(),abs() \*/  
 #include<process.h> /\* exit() \*/  
#include"stdlib.h"  
struct tnode  
{  
char data;  
struct tnode\*lchild;  
struct tnode\*rchild;  
};  
struct tnode tree;  
void createtree(struct tnode\*t)  
{  
struct tnode\*p=t;  
char check;  
if(p->lchild==NULL||p->rchild==NULL)  
{  
  printf("please enter the data:");  
      scanf("%c",&(p-data));  
  getchar();  
}  
if(p->lchild==NULL)  
{  
   printf("%c leftchild is null.Add?y/n\n",p->data);  
   scanf("%c",&check);  
   getchar();  
   if(check==&apos;y&apos;)  
   {  
     struct tnode\*q=(struct tnode\*)malloc(sizeof(struct tnode));  
 q->lchild=NULL;  
 q->rchild=NULL;  
 p->lchild=q;  
 createtree(q);  
   }  
}  
if(p->rchild==NULL)  
{  
  printf("%c rightchild is null.Add?y/n\n",p->data);  
   scanf("%c",&check);  
   getchar();  
   if(check==&apos;y&apos;)  
   {  
     struct tnode\*q=(struct tnode\*)malloc(sizeof(struct tnode));  
 q->lchild=NULL;  
 q->rchild=NULL;  
 p->rchild=q;  
 createtree(q);  
   }  
}  
}  
void preorder(struct tnode\*t);  
void inorder(struct tnode\*t);  
void postorder(struct tnode\*t);  
void main()  
{  
clrscr();  
tree.lchild=NULL;  
tree.rchild=NULL;  
createtree(&tree);  
preorder(&tree);  
printf("\n");  
inorder(&tree);  
printf("\n");  
postorder(&tree);  
}